XSS & CSRF

Computing

Two browser attacks constantly confused, and the defence for each.

XSS — injected script

- Attacker's JavaScript runs on your page
- Stored, reflected or DOM-based
- Steals tokens, rewrites the page
- Fix: escape on output, by context
- Fix: Content-Security-Policy header

CSRF — forged request

- A different site makes the user's browser act
- Uses cookies the browser sends automatically
- No script on your site required
- Fix: SameSite cookies
- Fix: anti-CSRF token per form

THE DIFFERENCE THAT MATTERS

XSS defeats every CSRF defence — once attacker script runs on your origin, it can read tokens. Fix XSS first.

> **Escape on output, not on input: the correct escaping depends on where the value lands — HTML, attribute, URL or JavaScript.**